



## RESEARCH ARTICLE

# Robust Resource Scheduling With Optimized Load Balancing Using Grasshopper Behavior Empowered Intuitionistic Fuzzy Clustering in Cloud Paradigm

G. Kiruthiga

PG and Research Department of Computer Science, Presidency College, Chennai, Tamil Nadu, India.  
g\_kiruthiga@yahoo.co.in

S. Mary Vennila

PG and Research Department of Computer Science, Presidency College, Chennai, Tamil Nadu, India.  
maryvennila13@gmail.com

Published online: 19 October 2020

**Abstract** – With the advancement in internet technology, everyone can able to utilize resources with low cost using cloud resources. There will be numerous requests for task scheduling to share resources in the cloud environment. When the task request is received by the cloud technology it should have the ability to distribute the workload among sharable resources in a balanced manner and effective utilization of resources. Machine learning and metaheuristic algorithms provide a dynamic part in balanced task assignments in the cloud paradigm. Existing unsupervised models-based load balancing, centroid selection is done randomly and imprecise job requests are not well handled by them. This paper aims to develop a clustering model-based task scheduling with the knowledge of behavioural inspired optimization algorithm in a highly balanced manner. A robust Intuitionistic Fuzzy C-means empowered grasshopper optimization has been anticipated in this work, which utilizes the merits of the Intuitionistic fuzzy and Grass Hopper algorithm for prominent task scheduling among virtual servers in a cloud environment. The results proved that IFCM-GOA reduces the makespan, execution time and, high balance load scheduling with improved cloud resource utilization.

**Index Terms** – Task Scheduling, Cloud Computing, Machine Learning, Intuitionistic Fuzzy C Means, Grasshopper Optimization.

## 1. INTRODUCTION

Cloud computing is the most emerging mechanism, providing a dynamic resource provisioning at higher levels of scalability. The cloud computing paradigm mainly focuses on the following key characteristics: providing self-service on-demand, pooling of resources, access to wide area network, and restrained acidity [1]. The allocation of resources is the toughest task in cloud computing since there is a gap between the number of requests for the services and the limited number of available resources. Hence, an effective mechanism is essential to evenly allocate the available

resources to the incoming requests. New methods have been proposed in resource allocation and provisioning, namely Roulette Wheel Selection (RWS) method, Priority-based Queue (PQ) scheduling, and Drip Irrigation based Resource Allocation (DRA) for improving the entire cloud service performance to cloud users.

Cloud Computing is also known as a model of distributed computational over a huge number of shared pools with virtualized computing resources [2]. Cloud computing signifies a vision of giving different kinds of services to internet users. The cloud computing architecture can be divided into different parts such as the back end and front end [3]. The back end has a vast amount of network of data centers with countless various kinds of data storage systems, system programs and different applications [4]. The front-end signifies application, for example, web browsers, organizations and different kinds of cloud users. This process is symbolically believed that the Cloud Service Providers (CSPs) almost have infinite storage capacity and computation power.

Depending on the need of customers, the resources are allocated and managed by the cloud service providers in an effective way [5]. The main activity of cloud resource management is task allocation and execution submitted by client users. There are two main processes in the cloud they are resource providing and arrangement.

The provisioning task is a method of discovering suitable resources for a given task purely depends on the quality of the service, needed by cloud clients. The resource scheduling process is related to mapping and execution of client job requests in cloud depending on the resources chosen for resource provision.

Load balancing aims to highly satisfy the cloud users by reducing the response time of tasks and optimized utilization

**RESEARCH ARTICLE**

of resources. The standard existing models work well for the homogenous type of virtual machines and don't consider the resource demands and lead to additional overhead when they scan the entire list of virtual machines. The clustering-based load balancing tackles well the heterogeneous environment, it meets the demand of resource and decreases the overhead of the screening process, as the virtual machines are clustered based on their abilities

The main objective of this paper is to perform optimized load balancing in the cloud environment, by precisely handling the heterogeneous type of Virtual machines and to satisfy the cloud users with quality-based services. The optimization is achieved by clustering both incoming requests and available virtual machines resources are clustered based on their resource availability. This research work contributes a metaheuristic-based clustering model even in a situation of impreciseness in assigning the incoming task request. The intuitionistic fuzzy clustering determines the similar task request and the available resources by representing them with the grad of membership, non-membership and indeterminacy. The cluster centroids are selected based on the grasshopper optimization instead of random selection, this also improves the load-balancing process in the cloud environment more effectively.

In Section 2 discusses related work, section 3 explains in detail about the proposed methodology of the load balancing in the cloud environment, section 4 deals with the results and discussion of the proposed work compared to the existing models of load balancing and finally, the paper concludes the finding of this work.

## 2. RELATED WORK

Pradeep and Jacob [6] reported a comparison analysis of the quality of service-based scheduling scheme with different types of task scheduling schemes with their advantages and disadvantages. The optimal Time algorithm is developed by Raju et al [7] which aims to increase productivity in task performance. It uses the concept of map-reduce based task scheduling scheme to diminish workloads makespan.

Ge et al [8] established a task scheduling scheme using a genetic algorithm in a cloud environment. They used the whole tasks in the job queue and resource assignment is done by considering the reduction of makespan to accomplish balanced scheduling among virtual machines.

Jang et al [9] devised a genetic algorithm for task scheduling which focuses on the merits of QoS and the cost benefits to providers in cloud computing

Qiang Guo et al [10] developed ACO for scheduling the task in a cloud environment. The algorithm utilized the parameters such as pheromone collection and updating of the fitness function to discover the best scheduling scheme. Their work

aims to produce better makespan, less the cost and maintain a balanced load in the cloud environment.

Zuo et al [11] anticipated a scheduling scheme using ACO model. They used the budget constraint to analyze the previous feedback on the quality of service. This characteristic avoids the pitfall of the local optima problem in ACO and then from the negative feedback.

Hongbo Liu et al [12] devised a particle swarm intelligence-oriented scheduling job. The performance of PSO is compared with simulated annealing and genetic algorithm

Srinivasa and Raveendran [13] designed an evolutionary algorithm-based resource scheduling using the genetic algorithm they proved that the performance of this model greatly suits than the batch queuing heuristic with the control parameters such as mutation and crossover rate that influence the impact of the effective solution.

Juan et al [14] designed a swarm intelligence-based task scheduling scheme to overwhelm issues of cloud network using a cost vector approach. It evaluates scheduling structures through cost and developed the model on input tasks and their needed QoS constraints. Though it is effective it leads to more complexity.

Krishnasamy [15] introduced a hybrid PSO based job scheduling algorithm whose aim is to reduce the average operation time with limited resource utilization.

Alkayal et al [16] in their work also used PSO with a multi-objective task assignment with ranking strategy. The requests are allocated to the virtual machine depending on the rank. Its performance resulted in less waiting time and system performance is high.

Rao et al [17] devised a Teaching-Learning Optimization algorithm which works under two phases with the phenomena of teaching-learning environment the resources are scheduled to produce a better result.

Dipesh et al [18] developed a clustering-based load balancing in the distributed environment of cloud computing to provide effective service delivery. This is a two-phase load balancing approach in the first phase involves in the cluster the distributed cloud data centers and the second client clustering assignment is performed to distribute the user's request uniformly.

Amer et al [19] introduced a dominant sequence clustering for task scheduling with weighted least connection to perform load balancing. The task of users are clustering with dominant sequence clustering and each task are ranking using Modified Heterogeneous Earliest Finish and virtual machines are clustered using means shift clustering to achieve better results.

**RESEARCH ARTICLE**

Malinen et al[20] in their work a balanced clustering k-means algorithm is used for assigning the task to the resources and they used the Hungarian algorithm it optimizes the mean square error for the given cluster size and they are maintained equally.

Geetha et al [21] performed a clustered based load balancing using fuzzy C means clustering which performs a reduced scanning process. The overhead involved in scanning the list of available virtual machines are clustered based on their abilities and the concern resource demands are highly satisfied.

**3. PROBLEM STATEMENT**

The existing models discuss so far works well for the homogenous type of virtual machines and don't consider the resource demands thus it leads to additional overhead when they scan the entire list of virtual machines while each request of tasks and it results in an imbalance load schedule. The proposed model utilizes the concept of clustering-based load balancing tackles well the heterogeneous environment, it meets the demand of resource and decreases the overhead of the screening process, as the virtual machines are clustered based on their abilities.

**4. Resource Scheduling With Optimized Load Balancing Using Grasshopper Behavior Based Intuitionistic Fuzzy Clustering**

This proposed work handles voluminous request of services are received by the cloud servers, depending on the type of resource requirements is analyzed and based on their consumption, they are grouped using an empowered unsupervised learning model. An enormous amount of job request arises as cloudlets to use the resources in the cloud. The cloud resources have to be distributed evenly among the cloudlets with a short period of time. Scheduling tasks effectively in a cloud environment is achieved by many optimized techniques. This proposed model constructs a behavioural inspired clustering model for achieving optimized task scheduling with effective resource utilization of clouds. A robust Intuitionistic Fuzzy C-means with grasshopper optimization was introduced in this research work, which utilizes the merits of the Intuitionistic fuzzy and Grass Hopper algorithm by integrating them to achieve a better quality of service in cloud computing.

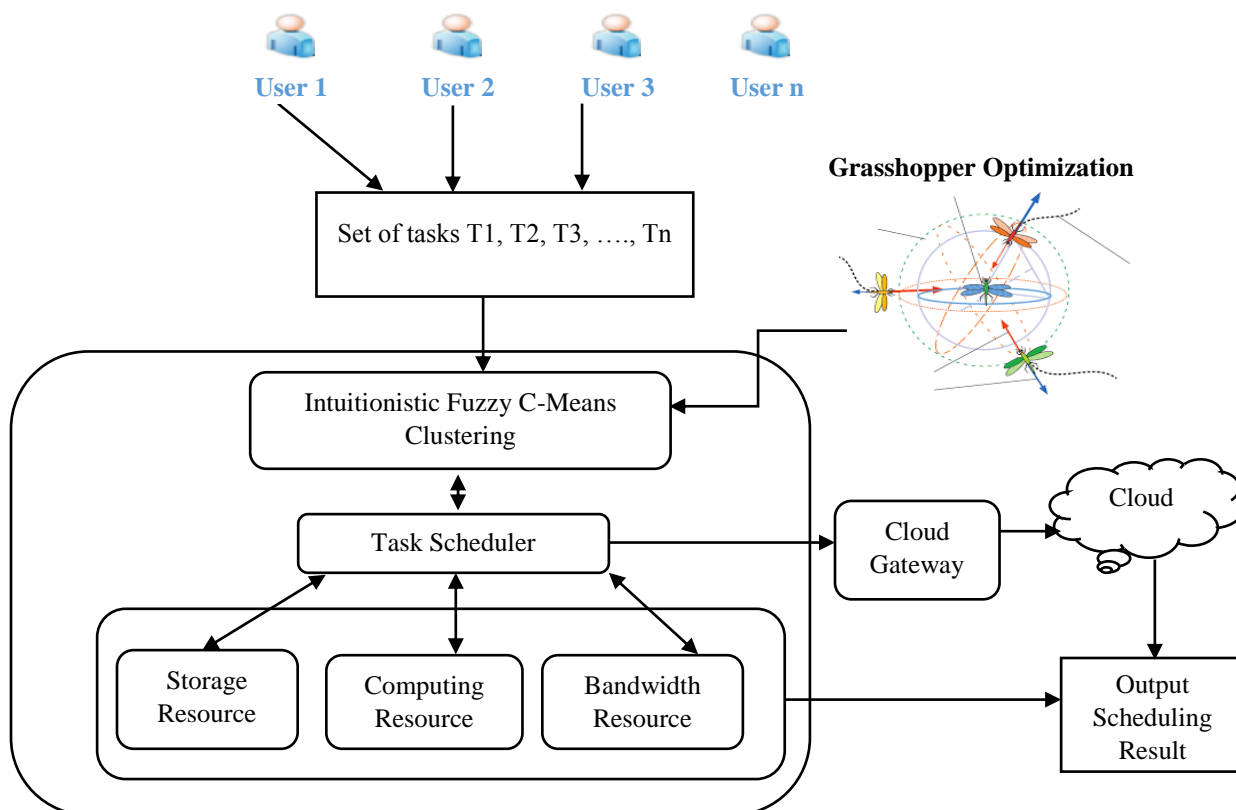


Figure 1 Overall Architecture of Grasshopper Behavior Empowered Intuitionistic Fuzzy Clustering in Cloud Paradigm

**RESEARCH ARTICLE**

In Figure 1, the incoming task scheduling request is clustered based on their need in storage, a resource to be involved and utilized and the bandwidth required for accomplishing tasks are computed. Likewise, the available virtual machines in a cloud environment are clustered based on their configuration and their availability of resources they are also clustered. The clustering is done by intuitionistic fuzzy C-Means where the parameters involved in clustering are represented in the intuitionistic fuzzy representation in terms of membership, non-membership and the hesitation degree.

The virtual machine or tasks has to be clustered by assigning some of them as centroids, to perform this grasshopper optimization performs the optimized centroid selection so that the clustering of similar virtual machines and the tasks are done effectively in the cloud environment. The load assignment among the virtual machines is evenly distributed using this strategy.

**4.1. Preamble of Intuitionistic Fuzzy Clustering**

The generalization of fuzzy theory is Intuitionistic Fuzzy theory which is developed by Atanassov [22]. In Fuzzy theory a set E is defined in terms of degree of membership  $\mu_E(x)$  whose value ranges between [0,1]. The Intuitionistic fuzzy represents the set E in terms of two different degrees they are membership  $\mu_E(x)$  and non-membership  $\nu_E(x)$ , which are independent to each other. The value of  $\mu_E(x)$  and  $\nu_E(x)$  lies between [0,1] with the constraint in equation 1 as,

$$\mu_E(x) + \nu_E(x) \leq 1 \quad (1)$$

Introducing grade of hesitation  $\pi$  in intuitionistic fuzzy significantly overcomes the issue of vagueness and impreciseness in determining the optimal resources for the incoming job requests in cloud environment more positively. Each incoming request parameters are measured in terms of  $\mu$  (membership),  $\nu$  (non-membership) and the  $\pi$  (hesitation). The value of  $\nu$  and  $\pi$  is obtained with the help of  $\mu$ . The non-membership of intuitionistic fuzzy [23] is calculated in equation 2 and as shown below,

$$\nu_B(y) = \frac{1 - \mu_B(y)}{1 + \beta \cdot \mu_B(y)}, \beta > 0 \quad (2)$$

Hesitation degree of intuitionistic fuzzy is articulated mentioned in equation 3 as follows,

$$\pi_B(y) = 1 - \mu_B(y) - \frac{1 - \mu_B(y)}{1 + \beta \cdot \mu_B(y)} | y \in Y \quad (3)$$

The hesitation degree plays a vital role in handling impreciseness in task scheduling for optimized assigned of virtual machines in cloud environment.

IFCM membership value is computed as shown in the below equation 4,

$$\mu_B^*(y) = \mu_B(y) + \pi_B(y) \quad (4)$$

To determine the objective function for the clustering process both the membership and hesitation degree are involved in determining the best centroids.

Intuitionistic fuzzy c-means objective function is signified in equation 5 and stated as,

$$OBJ(IFU, ct_1, \dots, ct_m) = \sum_{s=1}^m OBJ_s = \sum_{s=1}^m \sum_{j=1}^n (IFU_{sj}^*)^e \text{dst}(y_j, ct_s)^2 \quad (5)$$

**4.2. Procedure for Intuitionistic Fuzzy C-Means Clustering**

Input: Task Request TR = {tr1, tr2, tr3, ..., trn}

Output: Clustering similar Tasks

Begin

1. Assign cnas number of clusters
2. Set  $e > 1$  //degree of intuitionistic fuzziness
3. Set  $\beta > 0$  //Intuitionistic fuzzy Negation parameter
4. Initialize Intuitionistic fuzzy Matrix
5.  $U^{*(1)} = \{\mu_{ij}^{(1)}\}_{cn \times M} \forall i \in \{1, 2, \dots, cn\} \& \forall j \in \{1, 2, \dots, M\}$
6. Set  $L \leftarrow 1$
7. Update the cluster centers  $\vartheta_i^{IFS(L)} = \langle \mu_{\vartheta}(y_j)^{(L)}, \nu_{\vartheta}(y_j)^{(L)}, \pi_{\vartheta}(y_j)^{(L)} \rangle$
8. Compute  $\|Y_j^{IFS} - \vartheta_i^{IFS}\|$
9. Update Intuitionistic fuzzy partition matrix  $U^{(l+1)} = \{\mu_{ij}^{(l+1)}, \nu_{ij}^{(l+1)}, \pi_{ij}^{(l+1)}\}_{cn \times M}$
10. if  $\|U^{*(l+1)} - U^{*(l)}\| < \epsilon$  then  $\vartheta = \{\vartheta_i^{IFS}\}_{cn \times M} U^* = \{\mu_{ij}, \nu_{ij}, \pi_{ij}\}_{cn \times M}$
11. else Compute  $l \leftarrow l + 1$  go to step 13
12. Go to step 7
13. Stop the process

**Algorithm 1 Intuitionistic Fuzzy C-Means Clustering**

As presented in Algorithm 1, the number of task is assigned and checks the degree of intuitionistic fuzziness  $e > 1$  and negation parameter  $\beta > 0$ . Based on the values, initialize fuzzy matrix and update the cluster centers  $\vartheta_i^{IFS(L)}$ . Each time the fuzzy partition matrixes are updated ( $U^{(l+1)}$ ) and arrange the similar tasks.

**4.3. About Grasshopper Optimization Algorithm**

Grasshoppers are treated as hassle insects damage the crops in the agriculture field is known as pests. These insects may lead

**RESEARCH ARTICLE**

their life separately, but many times they form big swarms. For the farmers, it is a nightmare when the size of the swarm is too big. As adulthood and nymph, they have unique features of swarm behaviour. Nymph as a large population it moves like a rolling cylinder [24].

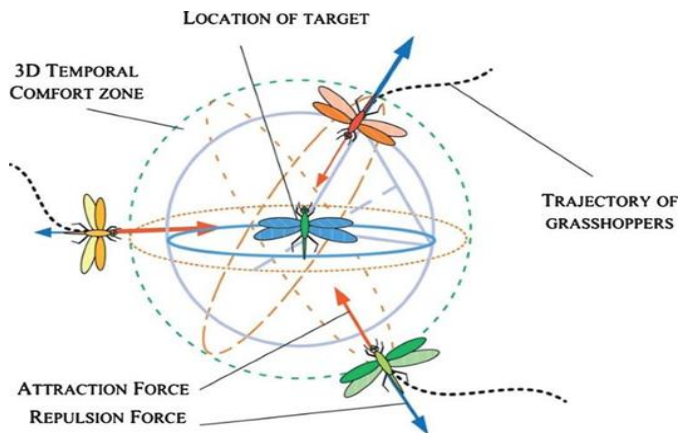


Figure 2 Overview of Grasshopper Behaviour Optimization

They eat most of the vegetation along their path during movement. To migrate along with distance, they frame swarm in the air [24]. During the larva phase, the swarm will move very slowly which is the noticeable characteristic of a grasshopper at this stage. But in adulthood their movement is abrupt. While they are in search of food, a swarm will be formed. From Figure 2, the behaviour is adapted in this paper to discover potential centroid to frame clusters for optimized resource scheduling in clouds. The artificial grasshopper optimization is comprised of a mathematical model of its food searching behaviour as depicted in the following algorithm.

To perform Intuitionistic Fuzzy C means clustering the initial centroids has to be selected, this work uses grasshopper optimization. The grasshopper model selects the best centroids by applying their food searching strategy with the help of the obtained fitness value of each virtual machine for the concerned task to be scheduled. The most appropriate virtual machine is selected by the grasshopper which has the highest fitness value and it is assigned to the task.

4.4. Algorithm for Grasshopper Optimization

1. Assign initial swarm value  $SWM_j = (j = 1, 2, 3, \dots, n)$
2. Assign  $c_{max}, c_{min}, noi$
3. Calculate each agent fitness value
4. Fit= Best (Search agent)
5. While ( $r < max\_iter$ )
  - 5.1. Update  $C' = C_{max} - r \frac{c_{max} - c_{min}}{noi}$
6. for every search agent

- 6.1 Apply normalization among artificial grasshoppers with their distance
- 6.2 Search Agents position is updated as follows,

$$SRA_i^z = \left( \sum_{j=i=1}^N c \frac{UB_z - LB_z}{2} sf(|sra_j^z - sra_i^z|) \frac{sra_j - sra_i}{dst_{ij}} \right) + Bst_z$$

(Where  $UB_z$  is the  $z^{th}$  dimension upper bound,  $LB_z$  is denoted as lower bound and  $Bst_z$  is the best solution recognized so far. The decreasing coefficient  $c$  refers to zone of attraction and repulsion)

- 6.3 If current search agent goes out of boundary region then set them back to the previous position

end {for}

7. Update Fit if there is better search agent found

7.1  $r = r+1$

end {while}

Algorithm 2 Grasshopper Optimization

Algorithm 2 initializes population of swarms, boundary for searching using  $c_{max}$  and  $c_{min}$  and the number of iterations. Each search agent position is updated based on the upper and lower dimension and the best fittest agent is considered in further iterations.

4.5. Cloud Resource Scheduling Using Intuitionistic Fuzzy Clustering Empowered by Grasshopper Optimization

Input:

Tasks Unallocated {Tsk}, Virtual Machines Unassigned {VM}

Output: Task Tsk Assignment to VMs // Makespan, Execution Time, Resource Utilization,

Procedure:

begin

1. for each  $task_{(i=1 \dots n)}$ 
  - 1.1. Priority-value =  $length(task_i) * priority(task_i) * deadline(task_i) * cost(task_i)$
- end for
2. Apply Intuitionistic Fuzzy Clustering and group them according to their similarity
3. Apply Grasshopper Optimization algorithm for selecting the cluster centroids

**RESEARCH ARTICLE**

```

4. for each VM{i...m}
    4.1 Determine VMi(features) //MIPS, bandwidth,
        memory, RAM capacity
    end for
5. Apply IFCM-GOA on VMs and group them as clusters
6. Divide the VM clusters and Tasks Clusters as low, medium
    and high priority types
7. for each task{i=1...n}
    7.1 Assign the concern VM depending on the cluster type
    end for
end {begin}
    
```

**Algorithm 3 Intuitionistic Fuzzy Clustering Empowered by Grasshopper Optimization**

As illustrated in algorithm 3, each task is prioritized based on their length, deadline and cost of each unallocated tasks, the best fittest virtual machine is selected by the grass hopper optimization and they are clustered as low, medium and high priority and the appropriate VM machines are assigned to each unallocated tasks based on their priority and availability of resources.

**5. RESULTS AND DISCUSSIONS**

This section discusses about the performance analysis of proposed Intuitionistic fuzzy Clustering empowered by grass hopper Optimization Algorithm (IFCM\_GOA) in resource allocation to the incoming tasks in cloud environment. IFCM\_GOA is simulated in java using cloudsim as cloud simulator. Datacentre used in this work is 5 and number of hosts under each datacentre is 2, totally 10 hosts are used and number of tasks/cloudlets ranges between 250- 1000. The metrics to assess the concert of the proposed model are makespan, resource utilization, energy consumption, degree of imbalance and execution time. The other scheduling models used for comparing the performance is k-means clustering [20], Fuzzy C-means (FCM) [21] clustering and conventional Intuitionistic fuzzy C Means (IFCM) clustering.

**5.1. Performance Comparison based on Makespan**

From the Figure 3 it is observed that the makespan of proposed model IFCM-GOA is very less compared to other three clustering models k-means, FCM and IFCM. The k-means algorithm only uses the predefined centroids which are selected in a random manner and started clustering the similar job request’s using the euclidean distance alone. In Fuzzy C means the belongings of a job request alone is considered for finding the similarity and its non-belongingness completely ignored in FCM. The IFCM considers both belongingness and non-belongingness of the job requests for each centroid, but

selection of centroid during each iteration is done only based on their distance and the initial centroid is selected in a random manner. Thus, with the help of grasshopper optimization the centroids are selected more prominently to cluster the most appropriate job request with the similar cluster.

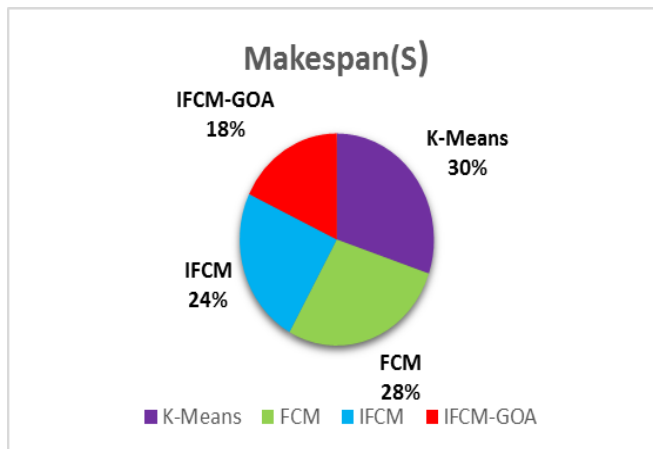


Figure 3 Comparative Analysis of Four Different Clustering Model Based on Makespan for Resource Scheduling in Cloud Environment

**5.2. Performance Comparison Based on Degree of Imbalance**

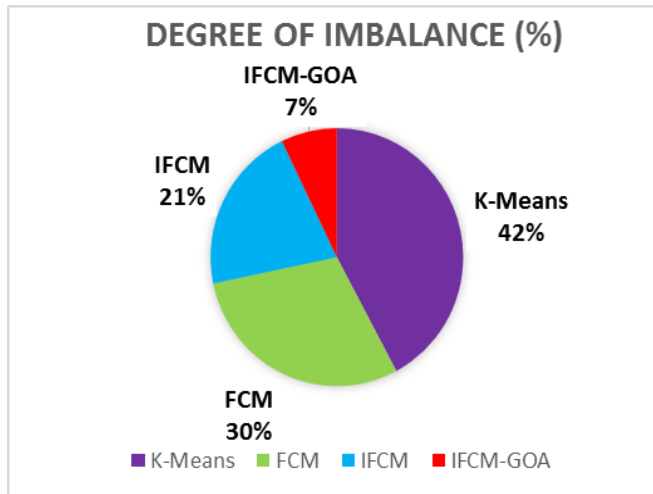


Figure 4 Comparative Analysis of Four Different Clustering Model Based on Degree of Imbalance during Resource Scheduling in Cloud Environment

Degree of imbalance is a measure of load imbalance between virtual machines in cloud paradigm and it is calculated as shown,

$$Deg\_imb = \frac{ET_{max} - ET_{min}}{ET_{avg}} \quad (6)$$

**RESEARCH ARTICLE**

Where  $ET_{max}$  is the maximum execution time of virtual machines,  $ET_{min}$  refers to minimum execution time of virtual machines and  $ET_{avg}$  is the average execution time of virtual machines.

Figure 4 portrays the measure of imbalance among virtual machines that are handled by the four different clustering models in the cloud environment. The proposed IFCM-GOA greatly balances the load with its knowledge of defining the hesitation degree and centroids are selected more intelligently by the searching behavior of the grasshopper. The load balancing of remaining clustering models only considered only the similarity of the job requests and the balancing among the virtual machines is not greatly focused on the existing models.

**5.3. Performance Comparison Based on Energy Consumption**

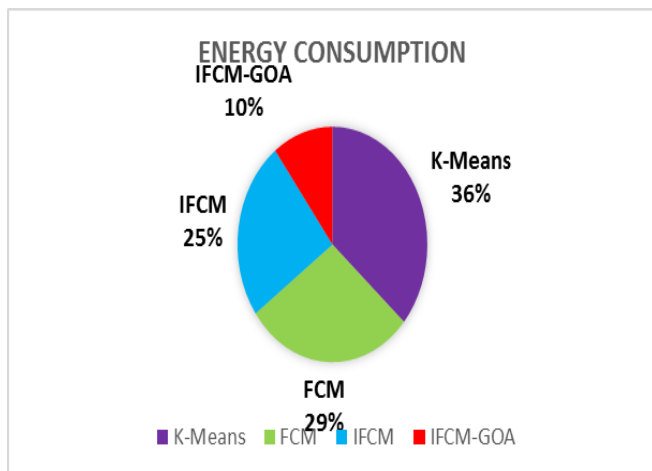


Figure 5 Comparative Analysis of Four Different Clustering Models Based on Energy Consumption during Resource Scheduling in Cloud Environment

Figure 5 shows the energy consumed by four different clustering models. The highest energy is consumed while using k-means based task scheduling in the cloud environment. This is because the centroid selection is arbitrary for all three existing models k-means, FCM and IFCM. The proposed model IFCM-GOA selects the initial cluster centroids and during each iteration of clustering by the metaheuristic model grasshopper optimization. With its food searching behavior the best fittest task scheduling is considered between incoming task requests and the available virtual machines. The resources are utilized in a balanced manner and thus the energy is consumed very less in a cloud environment which satisfies the cloud service provider's quality and the client's requirements.

**5.4. Performance Comparison Based on Resource Utilization**

The result of the resource utilization in the cloud environment is depicted in Figure 6 by applying the four different

clustering models. In an adversarial environment, impreciseness about the incoming job requests needs and selection of appropriate virtual machines to complete the assigned task are not fairly handled by the existing models. The Intuitionistic fuzzy clustering groups the tasks as heavy, medium, low, etc. The virtual machines are assigned to these tasks based on their requirement and depending on the availability of resources the incoming tasks are assigned to the virtual servers and thus achieved a higher rate of resource utilization compared to other clustering models.

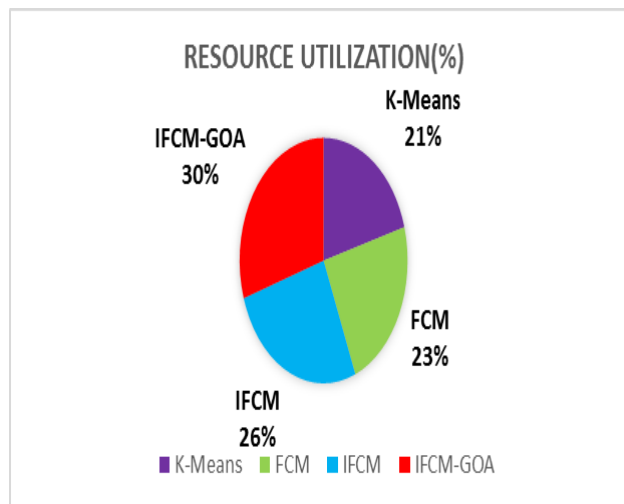


Figure 6 Comparative Analysis of Four Different Clustering Model Based on Resource Utilization during Resource Scheduling in Cloud Environment

**5.5. Performance Comparison Based on Execution Time**

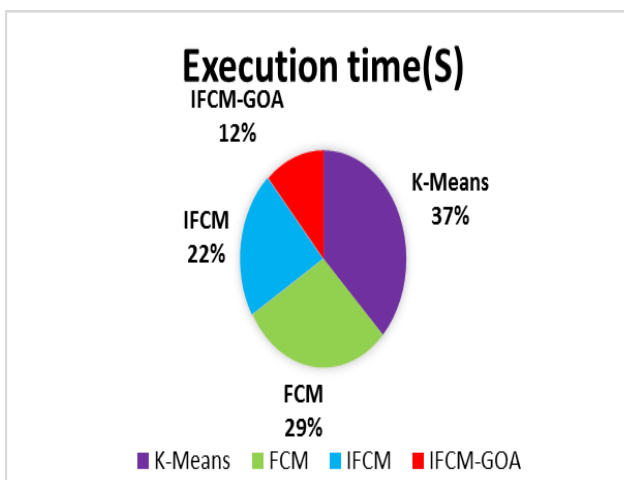


Figure 7 Comparative Analysis of Four Different Clustering Model's Execution Time during Resource Scheduling in Cloud Environment

The execution time performance of four different clustering models is depicted in Figure 7. The execution time of the



## RESEARCH ARTICLE

proposed IFCM-GOA is comparatively less while comparing with other clustering-based task scheduling policies in the cloud environment. The representation of each task is done depending on execution time, throughput, response time and turnaround time. The ability to represent each task based on the grade of membership and non-membership the clustering is carried out faster with the optimized cluster centroid selection, reassignment of centroids is greatly reduced and thus it consumes less execution time.

## 6. CONCLUSION

While there is impreciseness in determining the requirement of incoming cloudlets/ tasks in the Cloud environment, the effective scheduling of cloud resources will be the toughest challenge. The ultimate objective of this paper is to discover the similar resource request patterns and those similar tasks are clustered and the virtual machines available in the cloud are also clustered based on their characteristics by applying a novel clustering model known as intuitionistic fuzzy C means clustering. In this paper, additionally, the process of clustering itself is empowered by implying grasshopper optimization behavior to determine the centroids instead of selecting the initial centroid arbitrarily. The simulation results proved the prominence of the proposed IFCM-GOA plays a vital role in task allocation and cloud resource utilization more positively and precisely compared to the conventional K-means, Fuzzy C-Means and IFCM.

## REFERENCES

- [1] N. Kim, J. Cho and E. Seo, "Energy-credit scheduler: An energy-aware virtual machine scheduler for cloud systems", *Future Generation Computer Systems*, vol. 32, pp. 128-137, 2014.
- [2] S. Singh and I. Chana, "Consistency verification and quality assurance (CVQA) traceability framework for SaaS", 2013 3rd IEEE International Advance Computing Conference (IACC), Ghaziabad, pp. 1-6, 2013.
- [3] Li X. and Zheng M., "An Energy-Saving Load Balancing Method in Cloud Data Centers", In: *Frontier and Future Development of Information Technology in Medicine and Education. Lecture Notes in Electrical Engineering*, vol. 269. Springer, Dordrecht. [https://doi.org/10.1007/978-94-007-7618-0\\_35](https://doi.org/10.1007/978-94-007-7618-0_35).
- [4] F. Chen, J. Grundy, J.-G. Schneider, Y. Yang and Q. He, "Automated analysis of performance and energy consumption for cloud applications", In *Proceedings of the 5th ACM/SPEC international conference on Performance engineering*, ACM, pp. 39-50, 2014. <https://doi.org/10.1145/2568088.2568093>.
- [5] Tilak, S., and Patil, D., "A survey of various scheduling algorithms in cloud environment", *International Journal of Engineering Inventions*, vol. 1, no. 2, 36-39, 2012.
- [6] K. Pradeep and T. P. Jacob, "Comparative analysis of scheduling and load balancing algorithms in cloud environment", In: *Proc. of International Conf. on Control, Instrumentation, Communication and Computational Technologies*, pp. 526-531, 2016.
- [7] R. Raju, J. Amudhavel, M. Pavithra, S. Anuja and B. Abinaya, "A heuristic fault tolerant MapReduce framework for minimizing makespan in Hybrid Cloud Environment," *International Conference on Green Computing Communication and Electrical Engineering (ICGCCEE)*, Coimbatore, pp. 1-4, 2014. doi: 10.1109/ICGCCEE.2014.6922462.
- [8] Ge, Y., & Wei, G., "GA-based task scheduler for the cloud computing systems", *International Conference on Web Information Systems and Mining, WISM 2010*, Sanya, vol. 2, pp. 181-186, IEEE, 2010. doi: 10.1109/WISM.2010.87
- [9] Jang, S. H., Kim, T. Y., Kim, J. K., and Lee, J. S. "The study of genetic algorithm-based task scheduling for cloud computing", *International Journal of Control and Automation*, vol. 5, no. 4, pp. 157-162, 2012.
- [10] Guo Q, "Task scheduling based on ant colony optimization in cloud environment", In *AIP Conference Proceedings*, vol. 1834, no. 1, p. 040039, 2017.
- [11] Zuo, L., Shu, L., Dong, S., Zhu, C., and Hara, T., "A multi-objective optimization scheduling method based on the ant colony algorithm in cloud computing", pp. 2687-2699, 2015. doi: 10.1109/ACCESS.2015.2508940.
- [12] H. Liu, A. Abraham, A.E. Hassanien, "Scheduling Jobs on computational grids using a fuzzy particle swarm optimization algorithm", *Future Generation Computer Systems*, 2009.
- [13] Ch.Srinivasa Rao, B. Raveendra Babu, "DE Based Job Scheduling in Grid Environments", *Journal of Computer Networks*, vol. 1, no. 2, pp. 28-31, 2013.
- [14] Juan, W., Fei, L., and Aidong, C., "An Improved PSO based Task Scheduling Algorithm for Cloud Storage System", *Advances in Information Sciences and Service Sciences*, vol. 4, no. 18, pp. 465-471, 2012.
- [15] Krishnasamy K., "Task Scheduling Algorithm Based on Hybrid Particle Swarm Optimization In Cloud Computing Environment", *Journal of Theoretical and Applied Information Technology*, vol. 55, no.1, pp. 33-38, 2013.
- [16] Alkayal, E. S., Jennings, N. R., and Abulkhair, M. F., "Efficient task scheduling multi-objective particle swarm optimization in cloud computing", In *2016 IEEE 41st Conference on Local Computer Networks Workshops (LCN Workshops)*. pp. 17-24, IEEE, 2016. doi:10.1109/LCN.2016.024
- [17] Rao, R. V., Savsani, V. J., Vakharia, D. P., "Teaching-learning-based optimization: a novel method for constrained mechanical design optimization problems", *Computer-Aided Design*, vol. 43, no. 3, pp. 303-315, 2011.
- [18] Dipesh Pradhan, Feroz Zahid, "Data Center Clustering for Geographically Distributed Cloud Deployments, Primate Life Histories, Sex Roles, and Adaptability", pp. 1030-1040, 2018. doi: 10.1007/978-3-030-15035-8\_101.
- [19] Amer Al-Rahayfeh , Saleh Atiewi , Abdullah Abuhussein, MuderAlmiani, "Novel Approach to Task Scheduling and LoadBalancing Using the Dominant Sequence Clusteringand Mean Shift Clustering Algorithms", *Future Internet*, vol. 11,no. 109 , pp 1-15, 2019.
- [20] Malinen M.I., Fränti P. , "Balanced K-Means for Clustering. In: Fränti P., Brown G., Loog M., Escolano F., Pelillo M. (eds) *Structural, Syntactic, and Statistical Pattern Recognition*", *Lecture Notes in Computer Science*, vol 8621. Springer, Berlin, Heidelberg, 2014.
- [21] Geetha Megharaj, Dr. Mohan G. Kabadi, Rajani, Deepa M, "FCM-LB: Fuzzy C Means Cluster Based Load Balancing in Cloud", *International Journal of Innovative Research in Science, Engineering and Technology*, vol. 7, Special Issue 6, 2018.
- [22] Atanassov K, "Intuitionistic fuzzy logics as tools for evaluation of data mining processes", *Knowl-Based Syst*, vol. 80, pp. 122–130, 2015. doi:10.1016/j.knsys.2015.01.015.
- [23] Zeshui Xu, and Junjie Wu, "Intuitionistic fuzzy C-means clustering algorithms", *Journal of Systems Engineering and Electronics*, vol. 21, no. 4, pp.580–590, 2010. doi:10.3969/j.issn.1004-4132.2010.04.009.
- [24] Shahrzad Saremi, Seyedali Mirjalili, Andrew Lewis, "Grasshopper Optimisation Algorithm: Theory and application", *Advances in Engineering Software*, vol. 105, pp. 30-47, 2017.



**RESEARCH ARTICLE**

## Authors



**G. Kiruthiga** completed B.Sc. in Electronic Science (2001) from Bharathidasan University, Tiruchirappalli, India and Master of Computer Applications (2004) from SASTRA University, Thanjavur, India. She completed her M.Phil (Computer Science) in the year 2006 from Annamalai University, Chidambaram and currently pursuing Ph.D as Part-Time in Computer Science from Presidency College, Chennai. She is

currently working as an Associate Professor in Department of Computer Applications, Guru Nanak College, Chennai. Her areas of interest include Computer Networks, Cloud Computing and Bio inspired Algorithms.



**Dr. S. Mary Vennila** completed B.Sc. in Physics (1986) and M.Sc in Computer Science (1990) from Bharathidasan University, Tiruchirappalli, India. She obtained her M.Phil (Computer Science) in the year 2002 and Ph.D in Computer Science (2009) from Mother Teresa Women's University, Kodaikanal. She is currently working as an Associate Professor and Head of the PG and Research Department of Computer Science, Presidency College, Chennai, India. Her areas of interest include Networks, Grid

computing, Cloud security and Data mining.